Remarks

For the reasons set forth herein, Applicants request reconsideration and withdrawal of the obviousness rejection set forth in the final Office Action. Claims 1, 4-12, 15-34, 37-46, 51-60 & 63-72 remain pending.

Applicants gratefully acknowledge the indication of allowance of claims 8-11, 22, 30-33, 44, 56-59 & 70. This paper is directed to the remaining pending claims at issue.

Claims 1, 4-7, 12-21, 23-29, 34-43, 45-55, 60-69 & 71-72 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Ballard (U.S. Patent No. 6,078,960; hereinafter Ballard) in view of Colby et al. (U.S. Patent No. 6,006,264 A; hereinafter Colby). This rejection is respectfully, but most strenuously, traversed and reconsideration thereof is requested.

Applicants request reconsideration of the obviousness rejection on at least the following grounds: (1) the Office Action and purported combination of Ballard and Colby fails to state a *prima facie* case of obviousness against Applicants' claimed invention; (2) the Office Action misinterprets the teachings of Ballard and Colby in applying those teachings against Applicants' claimed invention, thus voiding the basis for the rejection; (3) the documents themselves lack any teaching, suggestion or incentive for their further modification as necessary to achieve Applicants' recited invention; and (4) the documents are not combinable as proposed in the Office Action.

Lack of *Prima Facie* Case of Obviousness:

Applicants' independent claims recite a particular technique for providing an ordered list of service addresses having certain characteristics. In each of Applicants' independent claims a distributed configuration manager of a computing environment is recited as performing certain functionality. Neither the Office Action, nor the applied references describe the existence of a distributed configuration manager of a computing environment *per se*, let alone the particular functionality accomplished by the manager in Applicants' recited invention. For this reason,

Applicants respectfully submit that the Office Action fails to state a *prima facie* case of obviousness against the claims at issue.

More particularly, Applicants recite "creating" *by a distributed configuration manager of a computing environment* a priority ordered list of service addresses to be used by a client node of the computing environment to reach a service of the computing environment. The particular component in Applicants' recited invention which creates the priority ordered list is a distributed configuration manager of the computing environment. No such distributed configuration manager is taught or suggested by Ballard or Colby, nor referenced in the Office Action. With respect to this language, the Office Action references (at page 3, line 9 thereof) FIGS. 4A, 4B, and column 6, lines 6-11 & 39-41 of Ballard. However, FIGS. 4A & 4B are a depiction of lists, while column 6, lines 6-11 & 39-41 of Ballard state:

> ... In some instances, the lists also include a respective load percentage for each of the listed ISP server computers. The percentage specified in the list for any given server computer may be the same or different than for the other servers identified in the list.
>
> For an embodiment in which there is a load percentage included in the list, such percentages are seeds for the server select function.

Nowhere in the recited material is a distributed configuration manager *per se* noted or depicted. As such, Applicants respectfully submit that the Office Action fails to state a *prima facie* case of obviousness against the particular claims at issue. Further, Ballard actually teaches away from the existence of a distributed configuration manager by teaching that the load balance list is provided by a system administrator. Specifically, Ballard teaches at column 2, lines 6-27:

> According to another aspect of the invention, a system administrator for the network servers storing the common data may update the load balance list. For example, in the event of a server computer failure, server computer maintenance or another reason, the system administrator can remove one or more server computer identifications from the load balance list and reapportion the load. The system administrator then downloads the updated load balance list to each operational server. When a client computer accesses one of the server computers and re-obtains a load balance list, the client computer receives the updated load balance list.

> According to another aspect of the invention, in the event a client computer determines that a server is non-responsive, such server is removed from the local load balance list of the client computer which made such determination. As a result, traffic from such client computer is automatically diverted to other servers that are functioning. Eventually when the client computer reobtains a load balance list from another server, an updated list will be available for the client computer, (assuming the system administrator independently has already determined that such non-responsive server is down).

Since the load balance list cited by the Examiner in the Office Action at column 6 as corresponding to Applicants' particularly recited ordered list of service addresses is provided by a system administrator, the Office Action, and the applied references, fail to state a *prima facie* case of obviousness against Applicants' claims. In Applicants' recited invention, a distributed configuration manager of the computing environment performs certain functions, including the creating of an ordered list of service addresses to be used by a client node of a computing environment to reach a service of a computing node. No such component and function thereof is taught by the applied references, or addressed in the Office Action.

Still further, since there is no distributed configuration manager in Ballard or Colby, Applicants respectfully submit that there can be no distributed configuration manager of a computing environment suggested or implied therein accomplishing the particular functions of the distributed configuration manager in Applicants' claims. For example, in Applicants' independent claims, the distributed configuration manager of the computing environment *creates* the ordered list of service addresses to be used by a client node of a computing environment to reach a service of the computing environment. Still further, Applicants' independent claims recite *providing* the ordered list created by the distributed configuration manager to the client node.

For at least the above-noted reasons, Applicants respectfully submit that the Office Action fails to state a *prima facie* case of obviousness against their independent claims, and as such, withdrawal of the rejection is respectfully requested.

Office Action Misinterprets the Teachings of Ballard & Colby:

As noted above, Applicants recite in their independent claims:

> *creating by a distributed configuration manager of a computing environment* a priority ordered list of service addresses to be used by a client node of a computing environment to reach a service of said computing environment, *said creating using a predefined equation to order a plurality of service addresses having the same ordering criterion, the predefined equation balancing use of said plurality of service addresses among said client node and at least one other client node of the computing environment*

Thus, in Applicants' recited invention, a distributed configuration manager of the computing environment creates a priority ordered list of service addresses. As noted above, Ballard fails to disclose the existence of a distributed configuration manager of the computing environment, or the creating of such a list by the distributed configuration manager. The load balance list referenced by the Examiner at column 6 is created by a system administrator and provided to the servers in Ballard. Clearly, Ballard thus teaches a distinct system and approach.

Still further, as emphasized above, Applicants recite that during the creating by the distributed configuration manager of the computing environment, a predefined equation is employed to order a plurality of service addresses having the same ordering criterion, the predefined equation balancing use of the plurality of service addresses among the client node and at least one other node of the computing environment. No such functionality is taught or suggested by Ballard. To the extent that this functionality is addressed in the Office Action, it is respectfully submitted that the Office Action misinterprets the teachings of Ballard. Specifically, the Office Action references column 5, lines 33-48 for an alleged teaching of such functionality at a distributed configuration manager of a computing environment. This conclusion is respectfully traversed.

Column 5, lines 24-48 of Ballard state:

> ... Another manner, as described in the background section, is to include a hardware load balancing device which responds to accesses for a given address. Multiple servers are coupled to the hardware load balancing device. A request from a client computer is routed to one of such servers through the load balancing device. In effect the server computers coupled to the load balancing device have the same address. Consider an example in which 10 server computers are coupled to a hardware load balancing device. Such device implements an algorithm to determine how to apportion the load (e.g., the requests to such common address) among the 10 computers. The apportionment may be a simple equal division (e.g., 10%), or may be according to some algorithm (e.g., round robin, random selection, a mathematical formula). These approaches are server side load balancing approaches, meaning the determination is made at the server side of the network.
>
> A percentage load for a given server computer may be based on the connect time to the given computer relative to the available connect time among all the computers whose load is being balanced. Alternatively, instead of relative connect time, load may be based upon a relative download volume, or a relative number of connections (regardless of length of connect time for each connection).

Clearly, the cited material in Ballard describes an alternative implementation wherein a hardware load balancing device is employed. In this alternate implementation described in the background of Ballard, all requests to a server address are routed through a common pipeline (i.e., the hardware load balancing device). (See column 1, lines 33-34 of Ballard.) Thus, the discussion at column 5, lines 33-48 is not related to the load balance list described further at column 6 of Ballard. Thus, the Office Action misinterprets Ballard in asserting that the load balance list thereof is created by a distributed configuration manager of a computing environment *using a predefined equation to order a plurality of service addresses having the same ordering criterion, the predefined equation balancing use of the plurality of service addresses among the client node and at least one other client node of the computing environment.* No such predefined equation is expressly or implicitly described in Ballard. Further, as noted above, the load balance list of Ballard is provided by a system administrator, with no discussion being set forth as to how the list is created by the system administrator. As such, Applicants respectfully request reconsideration and withdrawal of the obviousness rejection to the independent claims presented based upon the mischaracterization of Ballard, in particular, column 5, lines 33-48, in combination with the cited lines of column 6 regarding the load balance list thereof.

Yet further, Applicants' independent claims (e.g., claim 1) recite: *providing the ordered list created by the distributed configuration manager to the client node*. As noted above, there is no distributed configuration manager of the computing environment described in Ballard, Colby, or addressed in the Office Action. As such, the functionality recited cannot be express or implicit therein. In Ballard, the load balance list is obtained by a client upon access of a server computer. (See column 2, lines 14-16 & column 6, line 54 – column 7, line 9.) Clearly, the accessed server computer in Ballard would not be read by one of ordinary skill in the art as somehow equating to a distributed configuration manager of a computing environment having the facilities recited in Applicants' claims. As such, Applicants respectfully submit that the Office Action misinterprets the teachings of Ballard with respect to the functionality at issue.

For at least the above-noted additional reasons, Applicants request reconsideration and withdrawal of the obviousness rejection to the claims at issue.

Documents Lack any Teaching or Incentive for their Further Modification:

Applicants respectfully submit that a careful review of Ballard and Colby fails to uncover any teaching, suggestion or incentive for their further modification as necessary to achieve Applicants' recited invention. In this regard, neither patent teaches the existence of a distributed configuration manager for a computing environment. Further, neither patent teaches or implies the creation of a priority ordered list as characterized in the claims by such a distributed configuration manager of the computing environment. Still further, neither patent teaches the providing of a priority ordered list as characterized in the independent claims from a distributed configuration manager of the computing environment to the client node.

The consistent criterion for the determination of obviousness is whether the art would have suggested to one of ordinary skill in the art that the claimed invention should be carried out and would have a reasonable likelihood of success, viewed in light of the prior art. The suggestion and the expectation of success must be found in the prior art, not in Applicants' disclosure.

Applicants Traverse the Combinability of Ballard and Colby as Asserted:

Ballard describes client-side load-balancing in a client server network. In Ballard, load balancing is achieved at the client-side, rather than the server-side of a client-server network. Each client computer regularly receives a load balance list, enumerating respective addresses of multiple server computers. Each client computer executes a server selection function which determines the average load for each server in the list. In the event of a server computer failure, a system administrator can remove the server computer from the load balance list and reapportion the load. The client computer's list then is updated when the list is received during subsequent access. In the event a client computer determines that a server is non-responsive, such server is removed from the load balance list for the client computer which made such determination. (See Abstract of Ballard.)

Colby discloses a method and system for directing a flow between a client and a server. Specifically, a content-aware flow switch intercepts a client content request in an IP network, and transparently directs the content request to a best-fit server. The best-fit server is chosen based on the type of content requested, the quality of the service requirements implied by the content request, the degree of load on available servers, network congestion information, and the proximity of the client to available servers. The flow switch detects client-server flows based on the arrival of TCP SYNs and/or HTTP GETs from the client. The flow switch implicitly deduces the quality of service requirements of a flow based on the contents of the flow. The flow switch also provides the functionality of multiple physical web servers on a single web server in a way that is transparent to the client, through the use of virtual web hosts and flow pipes. (See Abstract of Colby.)

Applicants respectfully submit that one of ordinary skill in the art would not have extrapolated the teachings of Colby in the manner asserted in the Office Action, and combined those teachings with Ballard. Ballard clearly teaches functionality for client-side, load-balancing (see title thereof). In contrast, Colby teaches that "the entire process of server selection is transparent to the client". (See column 2, lines 57 & 58 of Colby.) Thus, it is respectfully submitted that one of ordinary skill in the art would not extrapolate an ordering concept from

Colby, which operates totally transparent to the client node, and then apply that teaching to the client node.

Further, Applicants respectfully submit that Ballard and Colby do not teach or suggest the creation of a priority ordered list having the characteristics recited in the independent claims *at a distributed configuration manager of the computing environment*. Thus, combining Colby with Ballard would still not equate to Applicants' recited invention. In Colby, the switching facility transparently redirects requests from one server to another server. The Office Action attempts to extrapolate this teaching and apply it to Ballard, wherein the "load balance list" obtained by the client from the server is the same load balance list (see column 2, lines 1 & 2 of Ballard) for multiple clients. This load balance list is not ordered in Ballard until the client computer receives the list. Ballard then teaches a process algorithm for the ordering of the list at the client node. This functionality is clearly contrary to the teachings of Colby, wherein the entire process of server selection is transparent to the client.

Still further, even if one were to combine the teachings of Colby with Ballard, it is submitted that the resultant ordering would occur at the client node, as expressly taught by Ballard. This is clearly contrary to Applicants' recited invention, wherein the distributed configuration manager of the computing environment creates the priority ordered list of service addresses, having the characterizations recited in the independent claims, and then provides the ordered list to the client node, which then uses the ordered list to reach the service. In Applicants' approach, the client node is alleviated from any functionality or processing requirements to create an ordered list in deciding how to reach a given service. In Applicants' approach, the client node simply uses the provided ordered list of service addresses.

Thus, because Colby and Ballard teach away from their combination as proposed in the Office Action, and because the resultant combination thereof still would not teach or suggest Applicants' claimed invention, Applicants respectfully request reconsideration and withdrawal of the obviousness rejection based thereon.

In summary, Applicants respectfully traverse the rejection of the independent claims based upon: the lack of a *prima facie* case of obviousness stated in the Office Action against

their invention; the misinterpretation of the teachings of Ballard and Colby; the lack of any teaching or incentive in the documents themselves for their further modification as necessary to achieve Applicants' recited invention; and the actual teaching away of the purported combination in the applied documents themselves.

For at least the above reasons, Applicants request reconsideration and withdrawal of the obviousness rejection to the independent claims at issue based upon the teachings of Ballard and Colby. The dependent claims are believed allowable for the same reasons as the independent claims, as well as for their own additional characterizations.

If a telephone conference would be of assistance in advancing prosecution of the subject application, Applicants' undersigned attorney invites the Examiner to telephone him at the number provided.

Respectfully submitted,

Kevin P. Radigan

Kevin P. Radigan
Attorney for Applicants
Registration No.: 31,789

Dated: January 27, 2006

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579